C-I-P
APPLIED CONCEPTS

ELECTRONIC AID FOR GAMES OF CHANCE

CROSS-REFERENCE TO RELATED APPLICATIONS

This application is a continuation-in-part of U.S. Patent Application Serial No. 08/709,221 filed September 3, 1996.

BACKGROUND OF THE INVENTION

1.0 Field of the Invention

This invention relates to the art of electronic games of chance and, more particularly, to a computerized Bingo aid that permits a Bingo player to store and play a plurality of games of chance, such as Bingo and instant win/lose lottery tickets.

2.0 Description of Related Art

As is well known, Bingo is a very popular game of chance wherein randomly selected numbers are called out in the sequence

of their selection by a game operator and players utilize cards on which are printed numbers corresponding to some of those which are called. The most common Bingo card is a five-by-five matrix containing numbered spaces in five vertical columns intersected by five horizontal rows. The centermost space is usually a free space and, hence, only 24 randomly selected numbers are printed on each card. After a number has been randomly selected and called out by an operator, each player places a marker or the like on the space on his Bingo card containing the called number if, in fact, it is present on his Bingo card. The players participate then in a game of chance to establish which will first obtain a series of numbers in a predetermined pattern, such as a straight line which is aligned vertically, horizontally or diagonally and calls out "Bingo."

Frequently, Bingo players play games on two different types of cards; namely hard cards and special cards. Hard cards are distributed on cardboard cards and are used repeatedly during play for usually playing straight Bingo games or double Bingo, meaning that a Bingo is won when the first player achieves a straight line either vertically, horizontally or diagonally. The special cards are distributed on sheets of paper to be played for only one game. These are typically used for special Bingo games which are

completed in patterns known as, X shape, picture frame shape, fill up shape, U-shape or C-shape and other special patterns all known in the art. It is known that some players are able to play thirty (30) or more cards at a time, while others have difficulty keeping up with the numbers called by the operator when playing more than four or five cards. These discrepancies are true whether the game is played with hard cards or special cards.

Devices that assist or aid in managing tasks associated with the Bingo are known and one such aid is disclosed in U.S. Patent 4,378,940 ('940) issued April 5, 1983, and herein incorporated by reference. The aid of the '940 patent while assisting a user during a Bingo game burdens the user by requiring him/her to use a card-reader to read the contents of a Bingo card to be played into the device. Further, the aid of the '940 patent is a stand-alone device not having any provisions for a communication link with another aid, such as, a computer to further assist or monitor the performance of the game of chance exemplified by Bingo. It is desired that an aid be provided that assists a player performing a game of chance, such as Bingo, but does not require a user to manipulate a card reader. Such an aid would also allow for a communication link with another system of the same invention or

computer systems. The systems would be used in managing the game or monitoring sales activity.

While playing Bingo, the player may also participate in a secondary game of chance that employs a form of lottery tickets. These tickets are also known as instants or pull-tabs and may be purchased from roaming vendors while the Bingo game is being played. The player rips open the ticket to instantly see if he/she has a predetermined pattern that indicates a winner. Such patterns include, but are not limited to, symbols, letters, numbers, words or phrases. The lucky player can then redeem the ticket upon finding one with a winning pattern. Bingo players encounter increased difficulty in playing their Bingo cards while playing these lottery tickets. It is desired that an aid be provided to assist a player of lottery tickets and to have this aid free of card reader manipulations, yet allow, if desired, the aid to be interconnected to a computer, via a communication link.

OBJECTS OF THE INVENTION

It is a primary object of the present invention to provide an aid for game of chances, such as Bingo and/or lottery tickets,

that is free of any user interaction with a card reader to read the contents of Bingo cards, and/or lottery tickets, into the aid itself before the game of chance can be initiated.

It is a further object of the present invention to provide an aid with computer capabilities for the performance of games of chances and having the provisions for establishing a communication path with another computer to assist in or monitor the performance of the game of chance.

It is another object of the present invention to provide a computerized Bingo aid that permits a Bingo player to store several Bingo cards and instant lottery tickets, thereby permitting a player to play a greater number of cards and lottery tickets at any one time than he/she might otherwise.

It is a still further object of the present invention to provide a computerized Bingo aid capable of communicating with a personal computer, another computerized Bingo aid, or a telephone modem.

It is a still further object of the present invention to provide a computerized Bingo aid that permits the selection of

various brands of prestored Bingo cards or Bingo cards that are manipulated by an algorithm to determine the win/lose status of the one or more games being conducted.

It is a still further object of the present invention to provide a computerized Bingo aid that permits the selection of an identification code stored in the computerized aid that identifies a plurality of brands of Bingo cards to be played, that resides on one board, sheet or any other form of media.

It is a still further object of the present invention to provide a computerized Bingo aid that permits the prestoring of the aforesaid identification code.

It is a still further object of the present invention to provide a computerized Bingo aid capable of selecting a plurality of both hard cards and special cards.

It is a still further object of the present invention to provide a computerized Bingo aid that permits the selection of various brands of prestored instant tickets.

It is a still further object of the present invention to provide a computerized Bingo aid that permits the player to have a prestored cash account within the aid that can be debited to pay for the Bingo cards and instant tickets.

It is a still further object of the present invention to provide such a computerized Bingo aid that permits the player to have a prestored cash account within said Bingo aid that can be credited whenever a game has been won on a Bingo card or instant ticket.

It is a still further object of the present invention to provide such a computerized Bingo aid that permits the selection of various prestored games or groups of Bingo games to be played in the Bingo cards.

It is a still further object of the present invention to provide a computerized Bingo aid capable of simultaneously playing a plurality of Bingo games on individual Bingo cards, each card being representative of one chance at winning a game.

It is a still further object of the present invention to provide a computerized Bingo aid capable of simultaneously playing

a plurality of Bingo games on a board, sheet or any other form of media, that is composed of a multiple number of individual Bingo cards. The board, sheet or any other form of media is in itself representative of one chance at winning a game.

It is a still further object of the present invention to provide a computerized Bingo aid that provides visual and audible indications to the player when a game has been won on a Bingo card.

It is a still further object of the present invention to provide such a computerized Bingo aid that provides visual and audible indications to the player when a game has been won on an instant ticket.

It is a still further object of the present invention to provide a computerized Bingo aid that permits the player to simultaneously play interrelated games on Bingo cards and instant tickets.

Another object of the present invention is to provide a computerized Bingo aid that is preprogrammed for a herein termed "prestored game schedule" which allows a player to play a session

of games without requiring the player to enter the sheet number of the bingo cards to be played along with the game pattern to be played for each and every game within the session of bingo that is to be played.

Furthermore, it is an object of the present invention to provide means to edit and update prestored game schedules.

SUMMARY OF THE INVENTION

The invention is directed to a programmable apparatus serving as an aid for assisting a player in performing games of chances and having prestored quantities that are accessed in response to player's entered quantities. In general, and as to be further described herein, the player identifies the brand of cards to be played, selects the cards to be played and then selects the games to be played. The player may also select a so called "schedule," having a brand of cards, the cards and games to be played having been previously selected and stored within the programmable apparatus.

The programmable apparatus comprises a processor, a readable memory, interactive means for providing an interactive dialogue for a player, and display means. The processor actions are directed by a computer program and has a plurality of ports. The readable memory stores a block of data representative of predetermined numbers and representative of predetermined patterns, wherein the predetermined numbers and patterns are correlated to each so as to be constructed into games of chance. The predetermined numbers and patterns are capable of being accessed and manipulated by the processor in response to the computer program. The interactive means is connected to one of the ports and provides an interactive dialogue between a player and the processor during the performance of the game of chance. The interactive means allows the user to enter data associated with the game of chance. The display means is connected to one of the ports and provides a display of the stored block of data, the data entered by way of said interactive means, and the visual status of the game of chance as it is being performed.

BRIEF DESCRIPTION OF THE DRAWINGS

Fig. 1 is a block diagram of the programmable apparatus of the present invention.

Figs. 2 - 19 are flow charts that cumulatively illustrate the sequence of operation of the programmable apparatus of the present invention.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

With reference to the drawing, wherein the same reference numbers indicate the same elements throughout, there is shown in Fig. 1 a block diagram of a programmable apparatus 10 of the present invention. The programmable apparatus 10 comprises a central processing unit (CPU) 12, commonly referred to as a processor, a readable memory 14, a keypad 16 serving as a control panel, and graphic display equipment 18.

In general, the programmable apparatus 10 provides assistance for a player, in an interactive manner, in the performance of a game of chance, such as Bingo or lottery tickets. The processor

12 has a plurality of ports and its actions is directed by a computer program encoded in the readable memory 14. The readable memory 14 has stored thereon a block of data representative of predetermined numbers and representative of predetermined patterns, wherein the predetermined numbers and patterns are correlated to each other to construct a game of chance, such as Bingo or lottery tickets. The control panel 16 serves as means, connected to one of the ports of the processor 12, for providing an interactive dialogue between a player, involved in a game of chance, and the processor 12 during the performance of the selected game of chance. The control panel 16 also allows the player to enter data associated with the game of chance into the processor 12 and to initiate and control the selected game of chance. The control panel 16 comprises a plurality of keys having the nomenclature shown in Fig. 1, but in addition thereto, the function keys F1, F2, F3 and F4 of Fig. 1 may be respectively identified with pictorial representations of a star, a pound symbol, a shamrock, and a heart. The graphic display 18 is connected to one of the ports of the processor and displays block of data, the data entered by way of the control panel 16, and selected events and data that may occur during the performance of the selected game of chance.

The processor 12 services the control panel 16 by means of a communication link comprising data paths 20 and 22 connected to one of the ports of the processor 12, and each of which operates with an interrupt request (IRQ) implementation, known in the art. The processor 12, via a communication link 24 connected to another of its ports services, in a parallel manner, the non-volatile R/W memory 14, a random access memory (RAM) 26, a read only memory (ROM) 28 having embedded therein a boot-strap routine, a real time clock 30, and the graphic display 18.

The processor 12 has an additional port 32 which accommodates a modem 34 that provides a communication path for the processor 12 to transfer and receive information from either a remote computer 36 or another system of the same invention. More particularly, the modem 34 provides a communication protocol between the processor 12 and another processor (not shown) situated in the remote computer 36. The readable memory 14 is capable of being programmed or re-programmed by computer 12 via the serial port 24 and, furthermore, the readable memory of the remote computer 36 is capable of being programmed or re-programmed by computer 12 via modem 34.

The processor 12 has a further port 38 connected to a security key 40, whose operation and servicing thereof by a computer program running in the processor 12 provides protection of stored data in the processor 12 or in any of the memory storage devices, such as readable memory 14, against unauthorized access.

The processor 12 has another port 42 that is connected to a piezoelectric element 44 and a visual indicator 46 so that upon determination by the computer program, the player may be provided with an audio and/or visual indication of a winning event occurring during the performance of a game of chance.

The processor 12 receives its excitation, via path 48, from a power supply 50 which also provides excitation to the graphic display 18 via path 52. The processor 12, in addition to having a real time clock 30 provided thereto, also provides for internal timing (in a manner known in the art) by the acceptance, via signal path 54, the output of a crystal 56 serving as an oscillator.

The programmable apparatus 10 is preferably a portable type and receives excitation from a portable power source 58 comprising

a charge circuit 58A, a battery pack 58B, and a power supply 58C. The charge circuit 58A is a battery charger circuit that receives energy at its input stage from an external power source 60 and provides a d.c. voltage at its output stage. The battery pack 58B comprises a plurality of chargeable batteries connected across the output stage of the battery charge circuit 58A and arranged to provide a cumulative voltage thereof. Once the batteries of the battery pack 58B are charged the charge circuit 58A may be removed from the external source 60. The power supply 58C has its input stage connected to and excited by the cumulative voltage of the battery pack 58B, and provides a plurality of d.c. voltages 62 at its output stage consisting of -17 volts (62A), +5 volts (62B), and +12 volts (62C) all of which are interconnected (not shown for the sake of clarity) to the elements shown in Fig. 1.

It is preferred that the readable memory 14 be a non-volatile memory, also known in the art as flash memory, such as that provided by an integrated circuit, but other readable memories are contemplated by the practice of the invention, such as floppy disks, CD-ROMs, hard drives, or any computer-readable storage memory, wherein, the computer program code is loaded into and executed by the processor 12, and the processor 12 becomes a primary element of the apparatus for practicing the present

invention. The present invention can also be embodied in the form of computer program code, for example, whether stored in a storage medium, loaded into and/or executed by a computer, or transmitted over some transmission medium, such as over electrically wiring or cables, through fiber optics, or via electromagnetic radiation, wherein, when the computer program code is loaded into and executed by a computer, the computer becomes an apparatus for practicing the invention. The communication link for such an application may be provided by the modem 34 of Fig. 1. When implemented on a general-purpose microprocessor, the computer program code segments may configure the microprocessor to create specific logic circuits. The computer program related to the Bingo and instant ticket games of chance of the present invention may be further described with reference to Figs. 2 through 17 which illustrate, by way of flow charts, the overall sequence of the operation of the present invention. The overall sequence is initiated by event 64 which is shown in Fig. 2.

OPERATION OF THE SYSTEM OF FIG 1

Event 64 shown in Fig. 2, occurs during power-up, and only at this time, the central processing unit (CPU) 12 resets and begins program execution and then passes control to program segment 66.

Program segment 66 indicates that the CPU 12 initiates and resets all of its elements of its hardware. The CPU 12 is an integrated device which has ancillary hardware that is fully programmable. The programmable hardware is programmed to function in a required manner. The hardware includes parallel input/output ports (I/O), serial I/O, interrupt system, analog to digital converters, system clock and memory mapping all known in the art and some of which are illustrated in Fig. 1. After initialization, program segment 66 passes control to program segment 68.

Program segment 68 indicates that the system of Fig. 1 is operated with two monitor programs that control all of the functions of the system. One program monitor controls the operations of a computerized bingo system, while the other program monitor is used to control the operations of a computerized instant ticket system. The first and second monitors are

downloaded from the nonvolatile read/write memory 14 to the system random access memory (RAM) 26. Running the monitors out of RAM 26 allows the system of Fig. 1 to conserve battery power and also allows various functions to be altered based on certain conditions which may arise. After downloading, program segment 68 passes control to program segment 70.

Program segment 70 creates the condition that in order to begin the game, scratch pad RAM must be cleared (i.e., set to all zeroes). The first and second monitor programs use many of these locations to keep count of various activities, or parameters, and to keep track of various conditions. For example, the parameters include the number of hard (HC) and special (SC) (i.e., paper cards) bingo cards that are currently in play, the number of instant tickets that can be drawn from, the number of winning bingo cards or instant tickets and loop counters, to name a few. In order words, any parameters or conditions that must be initially cleared are located in this area of control by program segment 70 which, when completed, passes control to program segment 72.

Program segment 72 is comprised of segments 72A and 72B which are related to displaying information. Program segment 70

provides for pictorial and text information to be displayed on the liquid crystal display (LCD) generally indicated in Fig. 1 as graphic display 18. The pictorial information is comprised of a series of 25 boxes that are grouped together to form a facsimile of a bingo card, another box showing information that has to be typed in from the keyboard, and a logo that represents the system of Fig. 1. The text information displayed on the graphic display 18 is in the form of a query. At this time, the player must convey to the system of Fig 1 the brand, type and number of bingo cards that comprise the board or sheet of paper that will be played on. The conveyance will be in the form of an identification code (see program segment 72B) that will be typed into the system utilizing the LOOK/DELETE key on the keyboard comprising the control panel 16 of Fig. 1. The system of Fig. 1 responds with a momentary text message in the LCD (graphic display 18) describing the type of cards that have been chosen. In some applications, the identification code is prestored as part of the first monitor program. After completion, program segment 72 passes control to program segment 74.

Program segment 74 indicates that the overall sequence of Figs. 2-17 is entering the game mode and interacts with a portion of working memory herein termed "match memory." Match memory is

cleared for both hard and paper cards. At the start of any game, these memory locations must be cleared in order to indicate that no matches presently exist on any cards. After completion, program segment 74 passes control to program segment 76.

Program segment 76 indicates that at this time, the system of Fig. 1 waits for a key closure on the keyboard on the control console 16. When a key is pressed, it will be detected and then decoded by program segment 76. After completion, program segment 76 passes control to program segment 78 composed of segments 78A and 78B.

Program segment 78 monitors for the occurrence of the LOOK/DELETE key of the control console 16. If the LOOK/DELETE key is pressed, the system of Fig. 1 again responds with a momentary message describing the type of cards that are currently in play. Also, the system of Fig. 1 tests the battery voltage and in turn places a picture of a battery in the LCD display (see program segment 78B). The picture graphically conveys to the player the level of charge that currently resides within the battery. It should be noted at this time that the system of Fig. 1 is always monitoring the battery voltage and informs the player when a lower battery condition arises. After completion, program segment 78

passes control to program segment 80, shown on Fig. 3, composed of segments 80A, 80B, 80C and 80D.

Program segment 80 determines if either of the HARD CARD (program segment 80A) or PAPER CARD (program segment 80C) keys are pressed, and, if so, the system of Fig. 1 respectively determines the serial numbers of the bingo cards (see program segments 80B and 80D) that have been chosen to be played, and then display them in the LCD (graphic display 18) display. Repeatedly pressing either of the aforesaid keys allows the player to scroll through and view all the bingo cards that have been chosen to be played for the respective key that was chosen. After completion, program segment 80 passes control to program segment 82 composed of segments 82A, 82B and 82C.

Program segment 82 determines if the PLAY key of the control console 16 is pressed, and, if so, the system of Fig. 1 is commanded to receive one or more sets of electronic instant tickets from a point of sale (POS) terminal via the serial channel input shown in Fig. 1 as communication link 32. The player would choose the brand and type of tickets to be loaded prior to the transfer via communication link 32. After completion, program

segment 82 passes control to program segment 84 composed of segments 84A, 84B, 84C and 84D.

Program segment 84 determines if a number key is pressed (0 through 9), and, if so, the current contents of a memory buffer within the working memory for the CPU 12 are shifted left one digit. The number corresponding to the pressed key is then inserted into the vacated right digit. The contents of the memory buffer is in turn displayed in the LCD (graphic display 18) display. After completion, program segment 84 passes control to program segment 86, shown in Fig. 4, composed of segments 86A, 86B and 86C.

Program segment 86 allows for various system functions to be called up by pressing the LOOK/DELETE key. Each called-up function is identified with a special code. Such functions include the following abilities and commands: a) ability to alter the aforesaid identification code (see program segment 78B) that identifies the type of cards that are being played; b) initiating a diagnostic test of the system; and c) the allowance of the entry of a special combination code that "unlocks" the system of Fig. 1. Each system is designed to be used a predetermined number of days or times. The system of Fig. 1 can be shut down after a period of

time if the user of the system of Fig. 1 has failed to reimburse the owner of the system of Fig. 1 for its use. The function provided by program segment 86 further includes the following abilities and commands: a) ability to alter the maximum number of bingo cards that can be played for any one game; b) ability to alter the maximum number of sequential bingo games that can be played; and c) initiating a programming of the system. The system of Fig. 1 may be designed to have all or part of its application programs and fixed data, including bingo cards, downloaded to it from a personal computer or another system of the same invention by way of the modem 34 of Fig. 1. This allows the system of Fig. 1 to enter a "learn" mode for the purpose of receiving information or a "teach" mode for the purpose of programming another system having a CPU, such as CPU 12.

All of the aforesaid functions associated with program segment 86 are not normally made available to the player, but are reserved for the proprietor of the system of Fig. 1, and the operator of the gaming hall wherein the system of Fig. 1 is normally used. After completion, program segment 86 passes control to program segment 88 composed of segments 88A, 88B, 88C, 88D and 88E.

Program segment 88 determines if either of the HARD CARD or PAPER CARD keys of the control console 16 is pressed, and, if so, the system of Fig. 1 respectively determines if the maximum number of cards have already been chosen to be played. If the maximum number of bingo cards have in fact already been chosen, the system responds with a momentary message informing the player of this fact. Program segment 88B or 88D, interacting as shown in Fig. 4, passes control to program segment 90.

Program segment 90 calculates the serial numbers of each bingo card that reside on the board or sheet for the type of bingo cards that have been previously chosen (see program segment 82). The serial number that was entered by the player is the basis on which the remaining serial numbers on the sheet are determined. After completion, program segment 90 passes control to program 92.

Program segment 92 operates in a manner determined by the type of card chosen to be played. More particularly, depending on the type of cards that have been chosen to be played, the serial number of a particular bingo card pertains to one of two types of cards. The first type is cards that have previously been prestored within the nonvolatile read/write memory 14 of the system of Fig 1. The other type is cards that are calculated with

an algorithm. The algorithm calculates a large library of bingo cards that contain nonrepeating permutations. The aforesaid serial number is now used to either find and read a prestored bingo card, or calculate one, and then store it within the system RAM 26 of Fig. 1. After completion, program segment 92 passes control to program segment 76 of Fig. 2 previously discussed. However, if program segment 92 was never entered, that is, program segment 88 of Fig. 4 passed control to program segment 94, then program segment 94 is performed.

Program segment 94 is composed of segments 94A, 94B and 94C and is performed if the system of Fig. 1 is designed to play a multi-faceted type of bingo game. Many bingo games are nothing more than the achieving of only one pattern of any of the cards. Others are comprised of multiple patterns in which any of the patterns can occur at any time on the cards. Play is continued until all of the patterns have been achieved. Still others are comprised of achieving a particular pattern(s) on a predetermined number of bingo cards on the same sheet or board Program segment 94 determines if either of the HARD GAME or PAPER GAME key is pressed, and, if so, the system of Fig. 1 utilizes the pattern number that has been typed in to identify the same pattern and type of game that is to be played for the respective key that was

chosen. For example, pattern #27 is a letter "X" game. If the player types in "00027" or "10027" via the keypad of the control console 16, the system of Fig. 1, more particularly the programs being run in CPU 12, understands this to mean that one bingo card is needed with the pattern "X" in order to be a winner. However, if the player conveys a pattern number of "40027" via the keypad of the control console 16, the system, more particularly the application programs being run in the CPU 12, will understand this to mean that four (4) bingo cards, on the same sheet, are needed with the pattern "X" in order to be a winner. In some applications, the game pattern numbers is prestored as part of the monitor in the form of a game schedule for the session of games to be played. If neither HARD GAME (segment 94B) nor PAPER GAME (segment 94A) key is depressed, program segment 94 passes control to program segment 84C (previously discussed with reference to Fig. 3); however, if either key HARD GAME OR PAPER GAME is depressed, program segment 94 passes control to program segment 96 of Fig. 5.

Program segment 96 indicates that the overall sequence of the programmable apparatus of Fig. 1 is now entering the play mode. The system of Fig. 1 now provides new pictorial and text information on the LCD display (graphic display 18). The

pictorial information still consists of a facsimile of a bingo card, a box that shows information that has been typed in from a keyboard, and the logo that represents the system of Fig. 1. However, the boxes within the bingo card will automatically contain the numbers on the bingo card that is closest to a winning pattern. Since no numbers have been played yet for the scenario hereinbefore described, the system of Fig. 1 defaults and shows the contents of the first bingo card that was chosen to be played.

Whenever numbers are played, any number on the displayed card that matches a number that was played are indicated, or "marked," with a reverse image of that same number. The text information conveys the serial number of the current card that is being displayed, the number of cards that is in play, the last three (3) numbers that were played and a list of all the numbers that are needed to win on any given pattern. After completion, program segment 96 passes control to program segment 98 composed of segments 98A, 98B and 98C.

Program segment 98, in particular segment 98A scans the keyboard of the control console 16 and program segment 98B determines if the LOOK/DELETE key is pressed, and, if so, the system of Fig. 1 tests the battery voltage (program segment 98C) and, in turn, places a picture of a battery in the LCD display

(graphic display 18). As before, the picture graphically conveys to the player the level of charge that currently resides within the battery. After completion,program segment 98 passes control to program segment 100 composed of segments 100A, 100B, and 100C.

Program segment 100 determines if either of the HARD CARD (segment 100A) or PAPER CARD (segment 100B) key is pressed, and, if so, the system of Fig. 1 respectively determines which cards are closest to a winning game pattern, and then starts to display (segment 100C) them. Repeatedly pressing either of the aforesaid keys allows the player to scroll through and view the contents of all the bingo cards that are being played for the respective key that was chosen. After completion, program segment 100 passes control to program segment 102 composed of segments 102A, 102B, and 102C.

Program segment 102, like program segment 100, determines if either of the HARD GAME (segment 102A) or PAPER GAME (segment 102B) key is pressed, and, if so, the system of Fig. 1 changes the contents of the LCD display (graphic display 18) and lists the name of each game pattern that is in play for the respective key that was chosen. After completion, program segment 102 passes control to program segment 104 composed of segments 104A and 104B.

Program segment 104 monitors for the pressing of the PLAY key which commands the system of Fig 1 to change the contents of the LCD display (graphic display 18) and list all of the numbers that have and haven't been played. After completion, program segment 104 passes control to program segment 106, shown in Fig. 6, composed of segments 106A, 106B, 106C and 106D.

Program segment 106 responds to function keys F1, F2, F3 and F4. The function keys (F1 through F4) are used to control the operation of the instant ticket side of the program means that is, the program being run in CPU 12. Each of the four function keys is further able to control a particular type of instant ticket that has been electronically loaded into the system. In other words, F1 allows a player to play on instant ticket #1, and F2 will allow a player to play on instant ticket #2, and so on. If either of keys F1, F2, F3 or F4 is depressed, program segment 106 passes control over to the operation sequence to be described with reference to Fig. 15, or conversely, if none of the keys F1, F2, F3 or F4 is depressed, program segment 106 passes control to program segment 108, shown in Fig. 7, composed of segments 108A, 108B, 108C and 108D.

Program segment 108 determines if a number key is pressed (0 through 9), and, if so, the current contents of a memory buffer (previously discussed) are shifted left one digit. The number corresponding to the pressed key is then inserted into the vacated right digit. The contents of the memory buffer is in turn displayed in the LCD display (graphic display 18). After completion, program segment 108 passes control to program segment 110 composed of segments 110A and 110B.

Program segment 110, in particular, segment 110A determines if the LOOK/DELETE key is pressed, and, if so, the system of Fig. 1 erases (program segment 110B) any indicated matches on the bingo cards for the number that was typed in. This in turn reflects on the contents of any displayed bingo card and on the numbers that were needed to win on any given pattern. After completion, program segment 110 passes control to program segment 112 composed of segments 112A, 112B, 112C, 112D and 112E.

Program segment 112 makes a distinction between the functions related to the HARD GAME and PAPER GAME keys. More particularly, at this juncture, the HARD GAME and PAPER GAME keys each has two different functions. If no number has been played yet, the pressing of either of these two keys informs the system of Fig. 1

that the number that has been typed in is an additional game pattern that must be played in addition to the one previously described. The system can simultaneously play a maximum of twelve (12) game patterns. If on the other hand, the game pattern entry mode is locked out, (i.e., a number has been played), the pressing of either of these two keys (HARD GAME or PAPER GAME) informs the system of Fig. 1 that the number that has been typed in is a game pattern that must be deleted. This situation would arise whenever another player has won on a particular game pattern and it is now necessary to remove the pattern that is no longer in play. As seen in Fig. 7, the program segment 112 has two exit paths, the first of which is back to program segment 98 (previously described) of Fig. 5 by way of segment 112D or 112E and the second path of which is via segment 112B to program segment 114, shown in Fig. 8, composed of segments 114A, 114B, 114C and 114D.

Program segment 114 determines if any of the function keys (F1 through F4) is pressed, and, if so, the system of Fig. 1 determines if any previously found progressive instant tickets contains the number that was typed in. As seen in Fig. 8, if any of the keys F1, F2, F3 or F4 is depressed, program segment 114 passes control to the operational sequence to be described with reference to Fig. 17, or conversely, if none of the keys F1, F2,

F3 or F4 is depressed, the program segment 114 passes control to program segment 116 composed of segments 116A, 116B, 116C, 116D and 116E.

Program segment 116 determines if the PLAY key is pressed, and, if so, the system of Fig. 1 electronically marks any bingo card that contains the number that was typed in. The number that was played is compared to all the numbers on all the bingo cards in the bingo card storage area of RAM 26. This area will be either for hard cards or paper cards depending on what type of cards is currently being played. If no match is found program segment 116 will prepare to compare the next number in the storage area of RAM 26 of Fig. 1. As seen in Fig. 8, program segment 116 passes control by way of first, second, or third paths, with the first path being by way of program segment 116A which passes control back to program segment 108C of Fig. 7 (previously described), the second path being by way of segment 116C to program segment 118, shown in Fig. 9, composed of segments 118A and 118B, and with the third path being by way of segment 116E to program segment 120 composed of segments 120A (see Fig. 8) and 120B of Fig. 9.

Program segment 118 determines if a match was found (see program 118A), and, if so, the memory location of the number with the match is stored in a table set up in an area of RAM 26. A register that is used in keeping count of the total number of matches is now incremented. If the second path related to program segment 118 was never entered, but rather the third path was followed then program segment 120 takes control from program segment 116.

Program segment 120 causes the advancement to the next number in the bingo card. More particularly, advancing to the next number in the bingo card storage area is accomplished by program segment 120 incrementing a memory pointer. The overall program being run in the CPU 12 will stay in this loop until all the numbers on all the bingo cards have been compared to the number that was played. If no matches were found, the program will be ready to accept new numbers to be played. As seen in Fig. 9, if no matches are found program segment 120B passes control to program segment 98A (previously discussed with reference to Fig. 5) and, conversely, if matches are found, program segment 120B passes control to program segment 122.

Program segment 122 operates so that the memory locations that were saved for every match are now used in "marking" that match on the proper bingo card. Some values or parameters must initially be set in order to begin marking the bingo cards. The scan card number (SCN) of program segment 122 must be set equal to one. The SCN value indicates which bingo card is currently being checked to see if the match is on it. A memory pointer is set to the area of match memory -3 for the type of cards (hard card/paper card) being played. Upon completion, program segment 122 passes control to program segment 124.

Program segment 124 fetches the location with a match from the table in the temporary storage area. Utilizing this location, the program segment 124 can determine the offset value for the number with the match. The offset value represents how far into the bingo card storage area the particular match was found. Upon completion, program segment 124 passes control to program segment 126 composed of segments 126A, 126B, and 126C.

Program segment 126 operates under the principle that every 24 locations in the bingo card storage area represent one bingo card. Therefore, the offset value can be used to determine which bingo card had the match, and where on that card the match

occurred. By continually reducing the offset value by 24, the SCN value can be updated to find the card with the match. The remaining offset value that is less than 24 indicates where on that particular card the match is located. As seen in Fig. 9, more particularly, as seen for segment 126C, when the offset value is less 24, program segment 126 passes control to program segment 128 composed of segments 128A and 128B.

Program segment 128 operates under the principle that once the SCN value has been determined, its corresponding area in match memory can be found. In match memory every bingo card is represented by three (3) memory locations. Each bit within these three locations represents a number on a bingo card or in other words, a location in the bingo card storage area. Hence, the SCN value can be used to determine the area in match memory that is used to represent a particular bingo card. Program segment 128 passes control to program segment 130 composed of segments 130A, 130B, 130C and 130D.

Program segment 130 operates under the principle that the final offset value indicates the location of a match on a bingo card. The final offset value can range anywhere from 1 through 24. The value of this offset corresponds to the location of a bit

in a card's match memory. As seen in Fig. 9, when the offset value is less than eight (8), program segment 130 passes control to program segment 132 composed of segments 132A, 132B and 132C.

Program segment 132 operates under the principle that once the bit location of the match is found in a card's match memory it will be set to indicate a match, or mark, on that card. Program segment 132 then determines if all the matches have been "marked" in the match memory of the bingo cards. If so, program segment 132 proceeds to check if any winning bingo patterns have occurred. Upon completion, that is, upon a YES decision from program segment 132B, program segment 132 passes control to program segment 134 of Fig. 10.

Program segment 134 operates in a manner so that before checking match memory for any possible winning bingo patterns, two parameters must first be set and which parameters are as follows: a) the number of bingo combinations needed to win a game and b) the number of different bingo combinations that have to be checked for winning combinations. Except for double and triple bingo, which need two and three respectively, all games require only one winning combination. Some forms of single, double and triple bingo may need varying amounts of combinations while the letter

"X" for example, only has one. Depending on which game is being played, the program must determine which values to use. Upon completion, program segment 134 passes control to program segment 136 composed of segments 136A and 136B.

Program segment 136 operates in a manner so that at this time, two memory pointers are set to their starting values. The memory pointers are the base address for match memory and the first address of a table in the program which holds all the possible winning bingo combinations for the game to be tested. These combinations are compared with the match memory of a bingo card to determine if the card has any possible winning combinations. Upon completion, program segment 136 passes control to program segment 138.

Program segment 138 consists of a loop counter which is loaded with a value of three. Program segment 138 must loop three times in order to compare the match memory for a bingo card against a particular bingo combination. Upon completion, program segment 138 passes control to program segment 140 composed of segments 140A, 140B, 140C, 140D, and 140E (see Fig. 11).

Program segment 140 operates under the principle that a bingo card's match memory and a bingo combination are both comprised of three bytes of data. A corresponding byte from each of them is now fetched from their respective areas in memory and then compared with each other. This comparison is accomplished in a two step process. First, the two bytes are ANDed with one another. This masks out any bits that are set in the byte from match memory that are not needed in comparing with the byte from the bingo combination. Secondly, the result is then exclusively ORed with the byte from the bingo combination. If the result was zero, the byte from match memory is the same as the byte from the bingo combination. A possible winning bingo combination may therefore be on this bingo card. If the result was nonzero, it means that the byte from match memory is missing one or more bits which in turn means that this bingo card is missing one or more matches. The program will exit the loop and determine if only one match is missing or more than one. As seen in Fig. 11, program segment 140, that is, segment 140E is exited by way of two paths (YES or NO answer to the decisional block 140E) and if a YES answer is yielded, it exits to program segment 142, but if a NO answer is yielded it examines the number of times the routine has been entered in a manner to be described.

Program segment 142 is composed of segments 142A, 142B, 142C, and 142D and operates on memory pointers. More particularly, the memory pointers that are used to access match memory and the bingo combination table are now incremented by program segment 142, while the loop count is decremented. If the overall program running in the CPU 12 has not gone through the loop three times, program segment 142 loops back and repeats the comparison loop. The loop must be completed three times in order to compare all of the bingo card's match memory to one of the bingo combinations. The last segment 142D determines if the loop count is zero and if NO, passes control to segment 140A of Fig. 10 (previously discussed), but if the answer is YES, segment 142D passes control to program segment 144 composed of segments 144A and 144B.

Program segment 144 indicates that the overall program reaches this point under one of two possible conditions. Either a winning bingo combination was found on the bingo or the card was found to be missing only one match. If a winning card was found, the winning card number is stored in a table and a counter that keeps track of the total number of winning cards found is incremented. However, if one number was found to be missing, the card will not be counted as a winner. The one number needed to generate a winning card, plus any others, will now be saved. As

seen in Fig. 11, program segment 144 passes control to the portion of the overall sequence of the present to be further described with reference to Fig. 12. As also seen in Fig. 11, and in a manner previously described, if program segment 140E yields a NO answer, then program segment 140E passes control to program segment 146 composed of segments 146A, 146B and 146C.

Program segment 146 operates under the principle that if the final result of the comparison between match memory and the combination table was nonzero, it means that for this bingo combination being tested, there is one or more matches missing from this byte. Before checking to see how many matches are missing, a flag is tested to see if this portion of the program has been entered earlier while testing the same match memory of a particular bingo card with the same bingo combination. The sole purpose of this routine is to seek out and find the one number that needs a match in order for this bingo card to have a winning combination. In order for this to be true, only one of the three bytes of match memory should be missing a match when compared one byte at a time with a particular bingo combination. It is possible to have one, two, or three bytes that are each missing one or more matches. Therefore, if this portion of the program is entered more than one time while comparing the match memory of a

bingo card with the same bingo combination, it will automatically mean that there are at least two matches missing on the card. If the program was found to have entered this routine twice, any previous number that was found to be missing a match will be scratched from a table that will contain these numbers. Upon completion, program segment 146 passes control to program segment 148.

Program segment 148 operates under the principle that if a byte from match memory does not match a byte from a combination table, the result will be tested to see if one or more matches are missing. Whenever a match is missing, a logic "1" will be present in a bit location of the byte. If there is only a single logic "1" the program will proceed to determine which number is missing a match. Otherwise, it means that there are at least two matches missing and the program will exit the routine. As seen in Fig. 11, if the answer to the decisional block of program segment 148 is YES, then program segment 148 passes control to program segment 150 composed of segments 150A, 150B, and 150C.

Program segment 150 examines an address in memory. More particularly, program segment 150 indicates the address of where the number, that is missing a match, is located in the bingo card

storage area. The address is calculated and then used to fetch the number from RAM 26 of Fig. 1. A table in memory contains all the numbers that need a match in order to generate a winning bingo combination. If the number read from RAM 26 is already stored in the table, it will not be stored again. If the number was not in the table, it will be stored and a counter will be incremented. As seen in Fig. 11, if the answer to the decisional block of program segment 148 was NO, then program segment 148 passes control to program segment 152 composed of segments 152A, 152B and 152C.

Program segment 152 operates under the principle that each type of game has a certain number of bingo combinations that must be checked against each card. The overall program being run in the CPU 12 will continually loop back until all the bingo combinations have been compared to the same bingo card. Before the next bingo combination can be checked, the memory pointer that is used to reference the bingo combination table is advanced to the next combination. As seen in Fig. 11, program segment 152 may be exited by two different paths, with the first path being by way of segment 152B that passes control to program segment 136B of Fig. 10 (previously discussed), and the second path being by way

of a YES answer to segment 152C which passes control to program segment 154.

Program segment 154 operates under the principle that most bingo games require that only one bingo combination appear on a card before a winner is declared. Others like double and triple bingo require two and three combinations respectively. At this time, program segment 154 determines how many bingo combinations are needed on a particular bingo card in order to win the game. Upon completion, program segment 154 passes control to program segment 156, shown in Fig. 12, composed of segments 156A, 156B, and 156C.

Program segment 156 decides whether to retain any or all of the numbers that were needed to generate a winning combination. The numbers would be retained if it was determined that zero or one bingo combinations were needed to win the game. However, in double bingo, or triple bingo, the program at one point will indicate that it needs two or more bingo combinations in order to win the game. When this occurs, the program must go back and delete those numbers that can generate only one possible winning bingo combination. Those numbers that can generate two or more winning bingo combinations will be retained as possible winning

numbers. Upon completion, program segment 156 passes control to program segment 158 composed of segments 158A, 158B and 158C.

Program segment 158 operates on the principle that a counter is incremented once all the bingo combinations have been compared to the match memory of a bingo card. This counter is used to keep count of all the cards that have been checked. This count will be compared to the number of cards that are stored in the bingo card storage area of RAM 26. If all the cards have not been checked, the program will prepare to loop back and check the next card. Once all the cards have been checked, the program will proceed to determine if any winning cards have been found. As seen in Fig. 12, program segment 158 passes control by way of segment 158C or 158B. Segment 158C passes control to program segment 136A (previously described with reference to Fig. 10), whereas program segment 158B passes control to program segment 160 composed of segments 160 and 160B.

Program segment 160 determines if any winning bingo cards were found by testing a counter. The value in the counter indicates how many winning cards were found. If the value in the counter is zero, the program will prepare to perform some tests on the numbers that were found to be needed in generating winning

bingo combinations. If none of the said numbers were found, the program will loop back to the beginning of the play mode where a new number can be entered and played. As seen in Fig. 12, program segment passes control by way of segments 160A and 160B, with segment 160A passing control to a portion of an overall program illustrated in Fig. 13 and with segment 160B passing control if a NO answer is yielded therefrom to program segment 98A of Fig. 5 (previously described), but if a YES answer is yielded from segment 160B, segment 160B passes control to program segment 162, shown in Fig. 13, composed of segments 162A, 162B, 162C, 162D and 162E.

Program segment 162 operates such that the numbers that were found to be needed in generating winning bingo patterns are collectively referred to as "set" numbers. These set numbers have been found to generate a winning combination on a card, but they may not be the final numbers that are needed to win a game. If a game is being played in which only one winning card is needed to win the game, then all of these set numbers are valid. But, if a game is being played in which more than one winning card must reside on a sheet to win the game, then some, or all of these set numbers may have to be deleted. True set numbers can only be valid if they can generate a winning condition. The only way to

test the validity of these numbers is to play the numbers and determine if a winning condition occurs. If one occurs, then the number is valid. If not, then the number must be deleted. Upon completion, program segment 162 passes control to program segment 164.

Program segment 164 causes the remaining set numbers, if any, to now be displayed to the player via the LCD display (graphic display 18). As seen in Fig. 13, program segment 164 passes control to program segment 98A of Fig. 5 (already described), and in a manner previously mentioned with reference to Fig. 12, program segment 166 of Fig. 13 receives control from program segment 160A.

Program segment 166 determines if a winning condition has actually arisen. Different games require different numbers of winning bingo cards to reside on a board or sheet of paper before a winner can be declared. A winner can be declared at this time if a game is being played in which only one winning card is needed. If more than one winning bingo card is needed, the program will loop back to the beginning of the play mode where a new number can be entered and played. As seen in Fig. 13, a NO answer yielded from the analysis of program segment 166 causes

program segment 166 to pass control to program segment 98C of Fig. 5 and, conversely, a YES answer causes program segment 166 to pass control program segment 168 composed of segments 168A, 168B, 168C, 168D and 168E of Fig. 13 and segments 168F and 168G of Fig. 14.

Program segment 168 operates on the principle that if it was determined that a winning condition has actually occurred, program segment 168 displays each of the winning bingo cards via the LCD display (graphic display 18). Each winning bingo card will have its serial number displayed along with all the numbers on the winning card. Any number that is found to have been "electronically marked" will be indicated as such with a reverse imaging of the number. Program segment 168 is commanded to display any other winning bingo cards that are contained on the winning board or sheet whenever the "LOOK/DELETE" key is pressed. Program segment 168 is further commanded to display any other winning boards or sheets whenever the "PAPER GAME" key is pressed. The "LOOK/DELETE" key is again utilized to display each individual winning bingo card on the board or sheet. Upon completion, that is when program segment 168F yields a NO answer, program segment 168 passes control to program segment 170 composed of segments 170A, 170B, 170C and 170D.

Program segment 170 operates on the principle that after all the winning bingo cards have been displayed, the program will wait until the player commands it to begin preparation for a new game. This process is initiated by pressing the "PAPER GAME" key of the control console 16 when there are no more winning boards or sheets to be displayed. Program control will now return to a game mode. Before the program enters the game mode, it will test to see if a paper game had just been played. If a paper game has just been played, the program will clear the paper card count to zero. In other words, the program will interpret this as meaning that there are no longer any paper cards stored in the bingo card storage area of RAM 26 of Fig. 1. The rules of bingo will only allow these cards to be played for one game only. New paper cards must be loaded for a new paper game. The program will then clear the LCD display (graphic display 18) and install new information into it as well as clear out program flags and any other parameters that must be set to zero in order to start a new game. Upon completion, program segment 170 of Fig. 14 passes control back to program segment 74 of Fig. 2 that clears the parameters and performs other house keeping functions needed to start the new game of bingo.

While playing Bingo, the player may also participate in a secondary game of chance that employs a form of lottery tickets. These tickets are also known as instants or pull-tabs. These tickets are normally purchased from roaming vendors while the Bingo game is being played. The player rips open the ticket to see if they have a predetermined pattern that indicates a winner. Such patterns include, but are not limited to, symbols, letters, numbers, words or phrases. The player can then redeem the ticket upon receiving one with a winning pattern. The system of Fig. 1 contains the same information that would be viewed on the ticket, but in electronic form. For users of the system of Fig. 1, it means that their ticket will be displayed to them pictorially via the graphic LCD display (graphic display 18). For the sake of simplicity, (but not for any limitation with regard to the scope of this invention) the explanation that is contained herein is for two types of instant tickets. Although, it is to be appreciated that the system of Fig. 1 is capable of playing a myriad of different types of instant tickets games. The operation of the electronic instant tickets may be described with reference back to Fig. 6 showing function keys F1, F2, F3, and F4.

The function keys (F1 through F4) are used to control the operation of the instant ticket side of the program means, that

is, the program running in the CPU 12 of Fig. 1. Each of the four function keys is further able to control a particular type of instant ticket that has been electronically loaded into the system. In other words, F1 will allow a player to play on instant ticket #1, and F2 will allow a player to play on instant ticket #2, and so on. As seen in Fig. 6, if any of the function keys F1, F2, F3 or F4 is depressed program segment 106 of Fig. 6 passes control to program segment 172, shown in Fig. 15, composed of segments 172A, 172B, 172C and 172D.

Program segment 172 determines if a function key (F1 through F4) is pressed, and, if so, the system of Fig. 1 initiates play on the selected instant ticket. The system of Fig. 1 exits this portion of the monitor if no tickets were entered, or none is left to be played, for the selected instant ticket. As seen in Fig. 15, if a NO answer is yielded from any of segments 172A, 172B, 172C, or 172D, control is passed back to program segment 98A of Fig. 5 already discussed, otherwise program segment 172 passes control to program segment 174 composed of segments 174A, 174B, 174C and 174D.

Program 174 operates so that the system of Fig. 1 randomly draws from the tickets that are available for the selected instant

ticket. For example, assume for the time being that the system of Fig. 1 operates on and includes a very common type of instant ticket for ticket #1 (program segment 174A). One form of the ticket is comprised of many different levels of winning tickets where each level is defined by the value of the winnings. It may allow one grand prize for obtaining three predetermined symbols and increasingly larger numbers of potentially winning tickets for lower prize values. These are sometimes referred to as "play backs." The reference is associated with the low prize value of such a winning ticket being redeemed for more instant tickets to be played. Further assume that the system of Fig. 1 operates on and includes a progressive type of ticket for ticket #2 (program segment 174B). A progressive type of ticket is one in which a predetermined number of tickets in a set will allow the bearer of a winning ticket to possibly collect a prize and continue playing on a different level for a larger grand prize. A progressive instant ticket can be comprised of many levels of play. Our example will assume to be playing on two levels. The grand prize winner may be defined by having a judge within the game draw another ticket that contains a number that identifies one of the bearers of the progressive tickets. Upon completion, program segment 174 passes control to program segment 176, shown in Fig. 16, composed of segments 176A and 176B.

Program segment 176 determines if the drawn ticket is a winner. A winning instant ticket is defined as any ticket that will offer any type of prize to the player, no matter the size of the prize. Whether a player "draws" a winning ticket or not, the ticket will always be displayed to the player via the LCD display (graphic display 18 of Fig. 1). Presentation of the ticket and its contents will serve as a confirmation for the player. As seen in Fig. 16, program segment 176A determines a winner and program segment 176 passes (if the answer is YES) control to program segment 178.

Program segment 178 determines the cash payout. More particularly, as determined by program segment 178, whenever a winning instant ticket is obtained, the system of Fig. 1, in particular program segment 178, determines the cash prize payout for the player. The size of the payout is based on many predetermined factors such as: a) the price to the player to purchase the ticket; b) the type of instant ticket being played; c) the odds of obtaining a card with a predetermined winning pattern; d) the number and amount of payouts to be made for the complete set of instant tickets that were sold; and e) the level of the prize that was obtained. Increasingly larger numbers of

potentially winning tickets are offered with lower payouts. As seen in Fig. 16, program segment 178 passes control to program segment 180.

Program segment 180 operates so that any payout that was won by the player may be credited to an electronic cash account within the system of Fig. 1 for the player. The player can then return to a point of sale (POS) terminal at a convenient time to receive his/her winnings or to have the account debited with the purchase of additional instant tickets. Upon completion, program segment 180 passes control to program segment 182 composed of segments 182A and 182B.

Program segment 182 determines if the winning ticket is of the progressive variety. If so, the serial number that is associated with the ticket (the serial number identifies the set from which the ticket was sold) will be saved. The ticket can now be used by the player in the next level of progressive play. The winning instant ticket will now be displayed to the player which is accomplished by program segment 182 passing control to program segment 176B which, in turn, passes control to program segment 184 composed of segments 184A and 184B.

Program segment 184 determines if the PLAY key is pressed, and, if so, the system of Fig. 1 then returns to playing bingo cards. Conversely, if a function key (F1 through F4) is pressed, (previously discussed with reference to Fig. 8) the overall program running in the CPU 12 of Fig. 1 sequence to program segment 186, shown in Fig. 17, composed of segments 186A, 186B, 186C and 186D.

Program segment 186 determines if a function key (F1 through F4) is pressed, and, if so, the system of Fig. 1 determines if any progressive tickets are still in play for the selected instant ticket. The system exits program segment 186 back to program segment 108C of Fig. 7 if no tickets are still in play. Conversely, if tickets are in play, program segment 186 passes control to program segment 188 composed of segments 188A, 188B, 188C and 188D.

Program segment 188 determines if any of the progressive tickets, that are still in play, contain the number that is currently being entered by the player. The number would represent a number that was imprinted on another ticket that was drawn by a judge, the sole purpose of which is to isolate one winning ticket. As seen in Fig. 17, upon completion program segment 188 passes

control to program segment 190 composed of segments 190A, 190B, and 190C.

Program segment 190 determines the payout to the player. More particularly, if a winning instant ticket is obtained, program segment 190 determines the cash prize payout for the player. As before, the size of the payout will be based on many predetermined factors such as: a) the price to the player to purchase the ticket; b) the type of instant ticket being played; c) the odds of obtaining the sole winning progressive ticket; and d) the number and amount of payouts to be made for the complete set of instant tickets that were sold. Any payout that was won by the player may now be credited to an electronic cash account within the system for the player. As before, the player can then return to a point of sale (POS) terminal at a convenient time to receive their winnings or to have the account debited with the purchase of new instant tickets. Whether a player obtains a winning ticket or not, the ticket will always be displayed to the player via the LCD display (graphic display 18). Presentation of the ticket and its contents will serve as a confirmation for the player. Upon completion, program segment 190 passes control back to program segment 184A of Fig. 15 which allows for the system of Fig. 1 to wait for the next play of the instant ticket.

In the overall operation, a player may enter the sheet number of bingo cards to be played along with the same pattern to be played for each and every game within the session of bingo that is to be played. Although such individual selection serves well its intended purpose, it is preferred that the programmable apparatus 10 of the present invention be fully preprogrammed with the brand, cut and collation of bingo paper to be used in a selected playing session along with the game patterns to be played throughout the session. The preprogramming for a particular session is referred to herein as a "schedule." The player may inform the programmable apparatus 10 by means of control panel 16 as to which schedule will be used and the identifying number for the book of collated bingo paper to be used with the schedule is then entered and the programmable apparatus 10 is ready to play any game in the session at any time.

It should now be appreciated that the practice of the present invention provides prestored data representative of predetermined numbers and prestored patterns both correlatable to the selected Bingo games and allows the Bingo games to be performed without the need of burdening a player with the manipulation of any card reader device.

It should be further appreciated, that the present invention computerizes the performance of a Bingo game, and provides for a communication link with another system of the same invention or another computer, via the modem 34 of Fig. 1, that allows for the performance of a Bingo game via any personal computer or any Bingo aid connected to the modem 34.

It should also be appreciated that the practice of the present invention provides for an aid in allowing the user to perform an instant game of chance characterized by a lottery ticket and having the benefits of computerized assistance.

The programmable apparatus 10 having the step-by-step sequences illustrated in Figs. 2-17 provided for a game of Bingo and that of a lottery ticket, may be merged so that both games are performed in a substantially simultaneous manner, and both games have their respective win/lose status displayed on the graphic display 18 of Fig. 1.

It should be further appreciated that the present invention prestores the contents of bingo cards that are played in prestored game schedules. The prestored game schedules allow for players to

select prestored bingo cards and game patterns to be played. More particularly, the present invention prestores information for bingo cards that are used in a given portion of the marketplace without having to deal with storing the information for every conceivable bingo card which is a difficult if not unachievable feat.

While not taking into account every conceivable bingo card, the present invention, in addition to prestored bingo cards of prestored game schedules, provides for the prestoring of preselected instant lottery style tickets that may be made available at a point-of-sale (POS) station. Further, the present invention provides the ability to set up prestored cash accounts via a POS station that can be debited to pay for bingo cards and instant tickets while also crediting these accounts whenever a payer wins.

It should be further appreciated, that the present invention provides the ability to edit prestored information concerning the bingo cards to be played within a given preprogrammed schedule. Said editing is accomplished via control panel 16 of the programmable apparatus 10.

It should be further appreciated, that the present invention provides for separate programming of the prestored schedules themselves. Without the benefits of this feature, the prestored schedule of information may be generated as a database embedded within the associated application software. Any desired changes in the prestored schedule would require changes to be made to the database which is a somewhat difficult task by non-technical personnel.

In general, the present invention provides a means to accept and change prestored schedules by technical or non-technical personnel through an application program being run on a personal computer. The user of said application program can compose up to 175 different schedules that are stored within one file that can be fed serially from the personal computer to the programmable apparatus 10 of the present invention. The newly programmed apparatus 10 can then be used to program further programmable apparatuses 10 of the present invention with the application software for the present invention and the schedules. The ability to change existing schedules, or to create new ones, allows the programmable apparatus of the present invention to accommodate any changes in bingo cards or bingo papers. The interactive dialogue between the technical or non-technical

personnel and the application program being run on a personal computer, hereinafter referred to as the schedule program, allows for editing existing prestored schedules or to create new ones. The schedule program is broken down into three main parts: configure schedules, convert schedules and program schedules.

The configure schedules allows the user of the present invention to assemble the order of games and define how the game of bingo will be played. The defined method of play is referred to as the schedule. The configure schedules portion uses standard and common nomenclature to allow the user to select options and fill in blanks creating a database for each game schedule entered. Several schedules can be configured together and programmed into the programmable apparatus 10 allowing the programmable apparatus 10 to be used during several different sessions without further reprogramming. The configure schedules are broken down into card parameters and game parameters. The card parameters allow for the type of bingo card to be played and how the bingo cards are collated into packets, or books, to be played by the player via the programmable apparatus 10. The game parameters allow for the selection of game patterns and parameters associated with each game. The information entered by

the user is entered into a database and saved to the memory of the personal computer.

The convert schedules convert or format the schedule program's database into a binary database readable by the programmable apparatus 10. The database is converted to a form that is collated and setup so the data can be used by the programmable apparatus 10. With this database in the programmable apparatus 10, the programmable apparatus 10 can transition from one game to the next without the user doing anything to setup the next game before starting. This capability eliminates confusion and the possibility the user would enter something that would make the programmable apparatus 10 play an unwanted game or play with unwanted parameters.

The program schedules read an ASCII text file generated from the convert schedules section and may program it into the programmable apparatus 10 by communicating through a serial connection between a standard PC and the programmable apparatus 10. Only the database needs to be programmed into the programmable apparatus 10 as the resident program is already in the programmable apparatus 10 and can interpret the information in the database to correctly play the next set of games correctly.

The schedule program of the present invention may be described with reference to Figs. 18 and 19 cumulatively illustrating a flow chart 192.

The flow chart 192 illustrating the scheduling program starts with the event (start) 194 of Fig. 18 which may be entered from the keyboard of the personal computer. The start event 194 passes control to program segment 196 (open database) which is a portion of the convert schedule. Program segment 196 reads all the variables and then passes control to program segment 198.

Program segment 198 then opens a programming file and saves all derived information to the new file which is used for programming. The first item saved to the programming file is the revision code which is used to verify the correct version for the resident program to be used within the programming apparatus 10 that in turn will further be using the converted programming file. The next item saved to the programming file is the number of schedules in the database followed by the number of games in each schedule. The handling of the scheduling parameters is accomplished by program segment 202. When program segment 202 is complete it passes control to program segment 204.

Program segment 204, as well as sequential program segments 206 and 210 (Fig. 19) make it convenient for the programmable apparatus 10 to access the data by providing an offset, or the number of memory locations, to the data for the first game in each schedule (segment 204). The offset is calculated and entered into the programming file followed by a calculated offset, or number of memory locations, to the beginning of each text message (segment 206). The user may define a text message, for each programmed schedule, to be displayed in the graphic display 18 of the programmable apparatus 10. After this information about the database is entered into the programming file, the information for each schedule is entered into the programming file. After all the information for each schedule (segment 210) is entered another offset to game one of each schedule is calculated and entered into the programming file followed by the game information. The last item to be entered into the programming file is the text messages (segment 214) to be displayed on the graphic display 18 of Fig. 1.

The program schedules of the schedule program 192 comprises program segments 208 (Fig. 18) and 212 and 214 (Fig. 19). The program schedules read the file generated in the convert schedules section (segments 198, 202, 204, 206, and 210) and communicates

the information to the programmable apparatus 10 by way of program segments 202, 208, 212 and 214. The programming file of segment 198 has one ASCII text number on each line and is normally 0 through 255. There can be numbers as large as 65535 however. When program segment 214 of Fig. 19 is complete it passes control to program segment 216.

Program segment 216 provides the housekeeping functions for closing out the programming file of program segment 198 and when program segment 216 is finished it passes control to event 218 which represents the end of the schedule program 192 of Figs. 18 and 19.

It is understood that the invention is not limited to the specific embodiments herein illustrated and described but may be otherwise without departing from the spirit and scope of the invention.